REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
22-02-2011	Technical Paper	FEB 2011 - MAR 2011

4. TITLE AND SUBTITLE
Fast Finish - New Techniques to Prevent NACK Feedback Implosion in Network Coded Multicast

5a. CONTRACT NUMBER
FA8720-05-C-0002

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Arman Rezaee, Linda Zeger, and Muriel Medard

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
MIT Lincoln Laboratory
244 Wood Street
Lexington, MA 02420

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
1777 N Kent St, Suite 9030
Arlington, VA 22209

10. SPONSOR/MONITOR'S ACRONYM(S)
OSD-ATL

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
We present a new feedback mechanism for wireless broadcast networks that utilize linear network coding. The setup considers transmission of packets from one source to n receivers over a single-hop erasure broadcast channel, but the proposed scheme is applicable to more general lossy networks. Previous feedback protocols can be roughly categorized as systems where all packets use ARQ which requires one feedback message from each receiver upon download completion. Our scheme utilizes a predictive model to ask for feedback only when necessary, i.e. if the probability that all receivers have completed decoding is high. In addition, the proposed NACK-based feedback mechanism enables n receivers to request the number of retransmissions needed for successful decoding within a single time slot. This is in contrast with traditional feedback systems whereby each user will be independently polled and thus the feedback presents a costly challenge to the performance of the network. In particular, we compare the performance of our mechanism with the state of the art negative feedback protocol NORM, [1]. We present simulation results that describe the implementation issues concerning the feedback time as well as analytical results that show the scalability of our proposal as the file size, number of receivers, and packet erasure probability increase.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF: U			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Zach Sweet
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	SAR	6	**19b. TELEPHONE NUMBER** *(include area code)* 781-981-5997

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

Fast Finish - New Techniques to Prevent NACK Feedback Implosion in Network Coded Multicast

Arman Rezaee
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, MA.
Email: armanr@mit.edu

Linda Zeger
MIT Lincoln Laboratory
Lexington, MA.
Email: zeger@ll.mit.edu

Muriel Médard
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, MA.
Email: medard@mit.edu

***Abstract*—We present a new feedback mechanism for wireless broadcast networks that utilize linear network coding. The setup considers transmission of packets from one source to n receivers over a single-hop erasure broadcast channel, but the proposed scheme is applicable to more general lossy networks. Previous feedback protocols can be roughly categorized as systems where all packets use ARQ which requires one feedback message from each receiver upon download completion. Our scheme utilizes a predictive model to ask for feedback only when necessary, i.e. if the probability that all receivers have completed decoding is high. In addition, the proposed NACK-based feedback mechanism enables n receivers to request the number of retransmissions needed for successful decoding within a *single* time slot. This is in contrast with traditional feedback systems whereby each user will be independently polled and thus the feedback presents a costly challenge to the performance of the network. In particular, we compare the performance of our mechanism with the state of the art negative feedback protocol *NORM*, [1]. We present simulation results that describe the implementation issues concerning the feedback time as well as analytical results that show the scalability of our proposal as the file size, number of receivers, and packet erasure probability increase.**

I. Introduction

Reliability is a challenging issue in wireless communications, particularly as the number of nodes becomes large, in which case conventional acknowledgment methods can result in unmanageable growth of feedback. We propose a new feedback mechanism for wireless broadcast networks that is built upon linear network coding. The novelties of our approach are that it provides a predictive model for the time at which transmissions are likely to be able to be terminated and it also reduces the feedback from all users to one time slot per request. The primary relevant piece of information the transmitter would derive from the feedback is the number of degrees of freedom missing at the worst receiver, and due to the use of network coding and the predictive model, this information is sufficient to substantially reduce the amount of feedback as well as unnecessary retransmissions by the transmitter.

Our proposed feedback mechanism has four main advantages over previous schemes: First, unnecessary initial polling by the transmitter is eliminated by use of the predictive model. Secondly, a significant reduction in the number of time slots allocated for feedback is achieved; this number currently scales with the number n of receivers, but under the new method will become a scalar of order 1. Thirdly, because of the significant reduction in the cost of feedback, the transmitter could poll the receivers more often, which allows for earlier termination of transmissions if all packets have been received. Fourthly, the mechanism can be implemented in a universal manner whereby its deployment will not require any changes in the physical layer.

One prime example of an appropriate application of this method can be seen in large latency and delay challenged networks described in [2], where feedback about received packets may be considerably delayed, reducing the feedback's usefulness and accuracy about the current state of the network. Another example is its use in any network with a large number of nodes that must receive the broadcast messages.

We study the performance gains of this feedback strategy, and compare it to the state of the art *negative* feedback protocols such as NACK-Oriented Reliable Multicast (NORM) introduced by Adamson et al. [1]. NORM protocol improves its predecessors by utilizing negative acknowledgments (NACK) instead of positive acknowledgments (ACK) which allows for scalability. NORM also uses end to end coding which equivalent to network coding in our set up is. We discuss the asymptotic throughput and delay performance of the network when transmission instances are modeled as discrete and continuous in time. We also show the robustness of this scheme to imperfections in the predictive model, including uncertainty in the number of receiving nodes n, the packet erasure probability p_e, as well as to losses of the feedback itself. In particular, we show that the number of time slots needed to reliably transmit k packets to n receivers with network coding scales as $\log n$ for large n and thus can be easily managed for a large network.

The rest of the paper is organized as follows: In Section II, the network model and parameters are introduced. In Section III-A, we evaluate the delay performance of the broadcast network under a discrete slotted model and provide heuristics about its behavior. In Section III-B, we derive the scaling laws for the system when a continuous model is assumed. In Section IV, we present the feedback mechanism through

This work is sponsored by the Office of the Secretary of Defense Contract FA8721-05-C-0002. Opinions, interpretations, recommendations, and conclusions are those of the authors and are not necessarily endorsed by the United States Government. Specifically, this work was supported by Information Systems of D,DR&E.

an example and give the general framework under which it can be implemented via CDMA codes. Finally, we provide a summary and concluding remarks in Section V.

II. Network Model and Parameters

Consider a wireless broadcast scenario in which a peer node transmits k packets to n independent users. In such systems a feedback mechanism is required to notify the transmitting node if all packets are received by the n users or further transmissions are needed. The transmitting node could be a base station or a peer node within the network, but for simplicity and ease of explanation we will call that node a base station. Let C_i denote the channel between the base station and the i^{th} user. A given channel C_i can be modeled as an erasure channel with parameter p_i, where p_i is the packet erasure probability on that channel. Assume that channels are independent across time and across receivers and the base station is interested in completing the transmission of its packets to all n users. We also assume that the base station uses network coding in the transmission of its packets, thus in the remainder of this paper we will use packets and degrees of freedom interchangeably.

In Sections III-A, and III-B, we assume that all channels are statistically identical and have the same packet erasure probability denoted by p_e. In Section IV we allow each channel to have an erasure probability p_i independent of other channels.

III. Predictive Model

The initial NORM protocol described by Adamson et al. requires *all* receivers to send feedback indicating their need for retransmissions. The result is a feedback traffic that grows linearly with the number of receivers and suffers in scalability. Many approaches have been suggested to reduce the amount of feedback, most notably the same authors have introduced a *NACK-suppression* scheme in [3] based on random back-off timers to delay retransmission requests. Despite the suppression scheme, the feedback traffic level rises slowly as the number of receivers increases, and the scheme does not offer a venue for prediction of appropriate feedback times. In this section, we will demonstrate the prediction capability of our feedback mechanism and show that the receivers will be polled if and only if there is a reasonable probability that they have completed their download. Figure 1 captures the difference in prediction capability of the two schemes by showing sample feedback times of both mechanisms. Note that with NORM, enhanced by the suppression scheme, NACKs occupy a proportion $\approx$ 10% of the slots throughout the transmission, which is in contrast to our scheme whereby the prediction allows for strategic placement of the NACKs at appropriate slots.

A. *Performance evaluation in a Discrete Model*

In this section, we analyze the number of time slots needed to reliably transmit a file of k packets to n receivers. The k coded packets could represent a file or image to be transmitted,

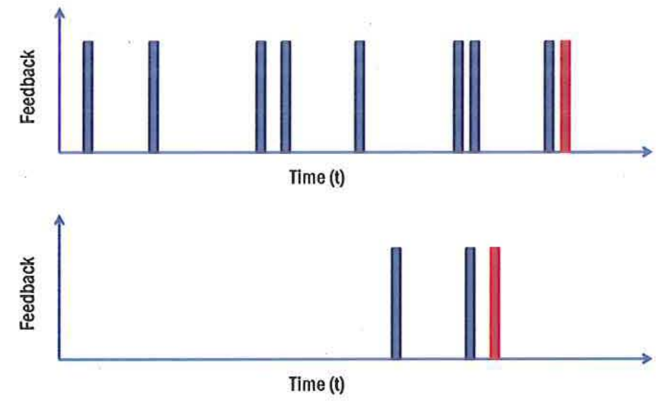

Fig. 1. Feedback times for NORM (top) vs. Fast Finish (bottom) when $n = 100$ and $p_e = 0.1$, the red bar denotes the end of transmissions.

for example. We will consider a slotted broadcast channel where each transmitted packet is received independently with probability $1 - p_e$ at any of the n receivers. Where p_e is the packet erasure probability. This is equivalent to n independent Bernoulli processes, all distributed with parameter $1 - p_e$, where we are interested in the shortest time until all processes have had k successful arrivals.

The transmission is completed when each of the n receivers has successfully received k or more coded packets. Let us denote the number of degrees of freedom (dof) missing at node i after t time slots by M_i^t, $M_i^t \in [0, k]$. We define another random variable $M_t = \max\{M_1^t, M_2^t, ...M_n^t\}$ to denote the number of dofs missing at the node that has experienced the highest number of erasures during t transmissions. The transmitter is expected to stop at $\{\min(t)|M_t = 0\}$. The probability that receiver i has received k or more coded packets in t time slots is:

$$Pr\{M_i^t = 0\} = 1 - \sum_{j=0}^{k-1} \binom{t}{j} p_e^{t-j}(1-p_e)^j \qquad (1)$$

Similarly, let us denote the probability that all n receivers have completed the download after t time slots by γ:

$$\begin{aligned}\gamma &= Pr\{M_t = 0\} = \left(Pr\left\{M_i^t = 0\right\}\right)^n \\ &= \left(1 - \sum_{j=0}^{k-1} \binom{t}{j} p_e^{t-j}(1-p_e)^j\right)^n \qquad (2)\end{aligned}$$

Note that γ is the probability that transmissions cease after t time slots. In the following 3 figures, we will show how γ changes as a function of p_e, k, and n. Figure 2 depicts γ vs. t for a range of erasure probabilities. Notice that the time at which transmissions can cease is very sensitive to packet erasure probability. This will be discussed in equation (11) where it is shown that completion time is inversely proportional to $(1 - p_e)$. As shown, for a network of $n = 1000$ nodes and $k = 10$ packets, $\gamma = 0.7$ is achieved after 21 time slots when the erasure probability is at $p_e = 0.2$. This number increases to 40 time slots when the erasure probability is at $p_e = 0.5$. Thus, in cases where the estimation of packet erasure probability is

inaccurate, it is better to have a feedback request earlier and avoid significant loss of throughput. Another important feature of this graph is the shape of the γ function; notice that the probabilities rise very sharply for smaller erasure probabilities than for larger probabilities.

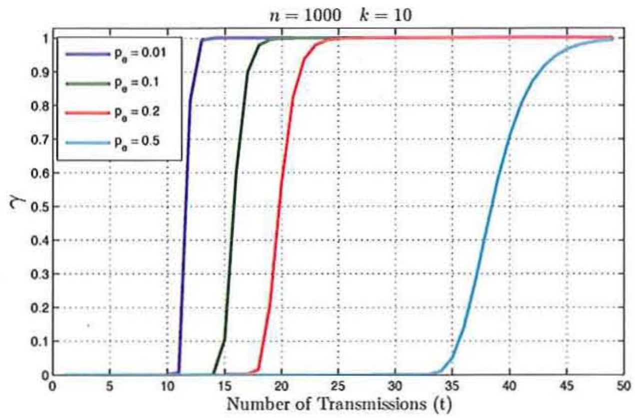

Fig. 2. Completion probability as a function of t for different erasure probabilities, p_e.

Figure 3, shows the dependence of γ on the size of the file, k. As shown, doubling the number of packets in the file will roughly double the number of transmissions needed for any given reliability. This is not surprising since the channel model is memoryless.

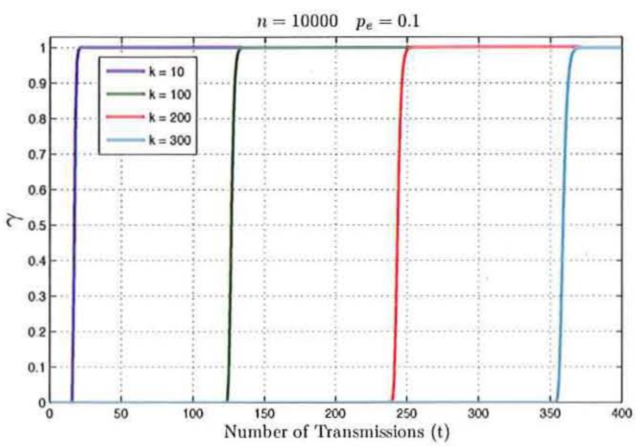

Fig. 3. Completion probability as a function of t for different values of k.

Figure 4 shows the performance of the system as we scale the number of receivers and shows the robustness of this transmission scheme to uncertainty in the number of receivers. We have plotted the completion probability γ when 10 packets are to be transmitted with $p_e = 0.1$ for networks ranging from 100 to 50000 nodes. As we will show analytically in Section III-B, the number of transmissions required for a given reliability increases logarithmically with the number of receivers n. Note that a reliability of $\gamma = 0.9$ is achieved after $16, 17, 18, 19$ transmissions for networks of $100, 1000, 10000, 50000$ nodes. The required number of transmissions for a particular reliability increases as $\log n$ as n increases.

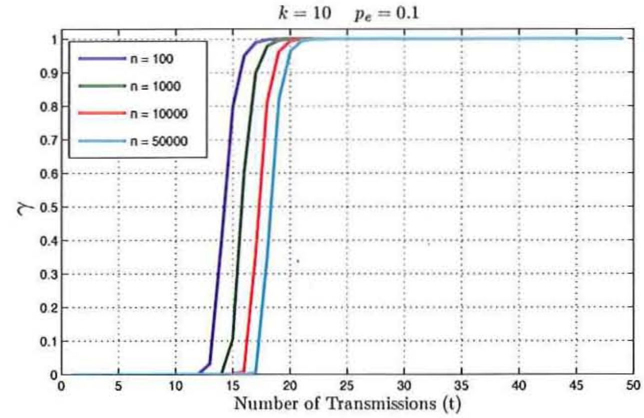

Fig. 4. Completion probability as a function of t for different values of n.

It is important to note that in all three figures, the CDFs have very sharp increases when erasure probability is not too high. In other words, a sharp increase in reliability is achieved by very few extra transmissions.

We can already foresee that combination of the prediction capability and the *single slot* characteristic of our scheme, which will be discussed in IV, allows us to avoid extra transmissions with an insignificant penalty of 1 time slot. This can be achieved by requesting a feedback when γ is small (on the order of 0.1) and the base station will be able to stop its extra transmissions γ proportion of the time.

Now, given that we have scheduled a feedback at time t we are interested in the number of extra transmissions that will be requested in the aforementioned feedback. In other words we are interested in the expected value of the minimum of n random variables.

Let us denote the number of nodes that have not completed the download at time t by a random variable N_1 and also use $\overline{N_1}$ as its expected value:

$$\overline{N_1} = E\left[\begin{array}{l}\text{\# of nodes that have received} \\ \text{less than } k \text{ dofs by time } t\end{array}\right]$$

Distribution of the number of nodes that have not completed the download by time t is as follows:

$$\begin{aligned} Pr\{N_1 = i\} &= \binom{n}{i}\left(Pr\left\{\begin{array}{c}\text{1 node completed} \\ \text{the download by t}\end{array}\right\}\right)^{n-i} \\ &\cdot \left(1 - Pr\left\{\begin{array}{c}\text{1 node completed} \\ \text{the download by t}\end{array}\right\}\right)^{i} \end{aligned} \quad (3)$$

since N_1 is non-negative, we have:

$$\overline{N_1} = \int_0^\infty 1 - F(x)dx \quad (4)$$

where F is the cumulative distribution function of N_1. Now given that we can find $\overline{N_1}$, we can find the expected number of transmissions needed to complete the download if the feedback slot was allocated at time t. It is important to note that during the feedback, the base station will know the number of packets missing at the worst receiver; in other words, the base station knows the maximum of $\overline{N_1}$ random variables.

Let us assume that the maximum number of packets missing is M_t and the transmitter will transmit $M_t^* = \frac{M_t}{1-p_e}$ packets. We can calculate the average probability γ^* that everyone has completed the download M_t^* time slots after the first feedback:

$$\begin{aligned}\gamma^* &= Pr\left\{\begin{array}{l}\text{everyone completed}\\ \text{the download by } t+M_t^*\end{array}\right\}\\ &= Pr\left\{\begin{array}{l}\overline{N_1} \text{ nodes completed}\\ \text{the download in } M_t^* \text{ slots}\end{array}\right\}\\ &\geq Pr\left\{\begin{array}{l}M_t \text{ packets are downloaded}\\ \text{at } \overline{N_1} \text{ nodes in } M_t^* \text{ slots}\end{array}\right\}\end{aligned} \quad (5)$$

Notice that equation (5) is the same as equation (2) if we replace k and t by M_t and M_t^* respectively. As a result, the sharp increase that was noticed in γ will also be present in γ^* and the entire file download will be accomplished with only a few feedback slots.

B. Performance evaluation in a Continuous Model

In this section, we will derive the scaling laws for the performance of the system when transmissions are modeled as continuous. Recall that the Poisson process is the continuous counterpart of a Bernoulli process. We model the arrivals at each receiver as a Poisson process and analyze the behavior of completion time as the number of receivers n grow.

Derivation:

Each of n users needs to receive k or more coded packets from a single transmitting node. In time t packet lengths, each of the n nodes *independently* receives a number of packets that is Poisson distributed, on the time scale of integral numbers of packet lengths, with parameter λt, where $\lambda = 1 - p_e$, and p_e is the packet erasure probability.

The probability that user i receives k or more coded packets within time t is thus:

$$Pr\{M_i^t = 0\} = 1 - \sum_{j=0}^{k-1} \frac{(\lambda t)^j \exp(-\lambda t)}{j!} \quad (6)$$

Hence the probability that all n users receive at least k coded packets in time t or earlier is (6) raised to the power of n. As in Section III-A we define γ to be the probability that *all* of the n users received k or more coded packets within time t. This probability γ, which is also the probability that the transmitter can stop sending coded packets, is:

$$\gamma = Pr\{M_t = 0\} = \left(1 - \sum_{j=0}^{k-1} \frac{(\lambda t)^j \exp(-\lambda t)}{j!}\right)^n \quad (7)$$

Rearranging terms yields

$$\lambda t = \ln\left(\sum_{j=0}^{k-1} \frac{(\lambda t)^j}{j!}\right) - \ln\left(1 - \gamma^{\frac{1}{n}}\right) \quad (8)$$

Expressing $\gamma^{\frac{1}{n}} = \exp\left(\frac{1}{n} \times \ln(\gamma)\right)$ allows for a Taylor series expansion in powers of $1/n$ for large n, since we consider a fixed γ, with $0 < \gamma < 1$. Thus for $|\ln(\gamma)| << n$,

$$\gamma^{\frac{1}{n}} \approx 1 + \frac{1}{n} \times \ln(\gamma). \quad (9)$$

It then follows that

$$-\ln\left(1 - \gamma^{\frac{1}{n}}\right) \approx \ln\left(\frac{n}{-\ln(\gamma)}\right) \quad (10)$$

Therefore, using (10) in (8) yields:

$$t \approx \frac{1}{\lambda}\left[\ln n - \ln\left(\ln\left(\frac{1}{\gamma}\right)\right) + \ln\left(\sum_{j=0}^{k-1} \frac{(\lambda t)^j}{j!}\right)\right] \quad (11)$$

At large values of time t, for $k << n$, the last term in (11) is small compared to the term linear in t on the left hand side. It is then seen that the time at which all n nodes have received all k packets with a specified high probability scales logarithmically with n. In particular, for a large probability $\exp(-n) << \gamma$ that the transmitter can stop sending coded packets at time t, we obtain:

$$t \approx \ln n. \quad (12)$$

We note that this termination time t is independent of the file size k, when coding is used.The $\ln(\ln(\cdot))$ dependence of t on γ shows that the termination time t defined by γ has a very weak dependence on γ, as seen in Figures 2 to 4. Thus once there is a significant non-zero probability that all packets are received by all receivers, not many more transmissions will be needed unless p_e is large, which renders the multiplicative factor $\frac{1}{\lambda}$ in (11) large.

According to [4], we can approximate the expected value of the minimum of n i.i.d. Poisson random variables with mean λ by:

$$\begin{aligned}E\left[\begin{array}{l}\text{min of n}\\ \text{Poisson rvs}\end{array}\right] &= \lambda + \lambda^{\frac{1}{2}} E[X_{1,n}] + \frac{1}{6}\left(E[X_{1,n}^2] - 1\right)\\ &+ O\left(\lambda^{\frac{-1}{2}}\right)\end{aligned} \quad (13)$$

where $X_{1,n}$ is the minimum of n standard normal random variables. Tables and asymptotic formulas for the expected value of $X_{1,n}$ can be found in [5] by Harter and in [6] by Sarhan and Greenberg. To see a more comprehensive treatment of this issue the authors refer you to [7], and [8]. Briggs *et al.* in [9] give an accurate account of the problems with a closed form expression that pertains to maximum/minimum of n Poisson random variables and they accurately estimate the maximum of n Poisson random variables. The keynote of their method in calculating the asymptotic behavior of the maximum of n Poisson random variables is that as n gets really large the CDF of the maximum will have two peaks which are 1 unit apart and this allows for an accurate approximation of the CDF. They use this fact to find the most probable value of the minimum of n Poisson random variables. We can use this value as the point at which we allocate the feedback slot.

Despite the recent work in [9], optimizing the time for feedback requires further knowledge and approximations regarding the expected value of the minimum of n Poisson random variables which is currently under investigation.

IV. Feedback Mechanism

We will introduce the mechanism in subsection IV-A through a specific example of a CDMA code, that of using jitter (pulse position indication), as a means by which nodes communicate to the base station the number of packets that must be retransmitted using only a single shared time slot. In subsection IV-B we will generalize various aspects of this example and discusses the range of design possibilities. Finally, in subsection IV-C, we show that jitter can be generalized to a range of code selection methods that indicate the amount of information the base station has to retransmit by utilizing the broad class of CDMA codes.

A. Example of Feedback Mechanism: Jitter

Given that the base station has k packets to deliver to n receivers, a single feedback slot is allocated after t transmissions of the base station, where t is chosen according to the prediction method described in III. During this slot, any of the n users that have not correctly decoded all k packets will send a short pulse to the base station; the timing of which indicates how many new degrees of freedom (dof) the base station needs to transmit for this user to recover all its missed packets. As before, let $M_i^t \in [0, k]$ denote the number of degrees of freedom missing at node i after t time slots. Within the feedback slot the time at which node i transmits its feedback pulse is determined by the realization (value) of M_i^t. Thus, we can think of the feedback slot as concatenation of subslots whereby the presence of a pulse in a specific subslot will indicate that a corresponding predetermined number or percentage of dof is needed. The exact method of correspondence between transmission in a specific subslot and number of degrees of freedom requested will be discussed in subsection IV-B.

It is assumed that network coding is used on the retransmissions of erased packets, since the total time to complete transmissions will be greatly reduced with network coding [10]. Therefore, only the number of packets a user did not receive needs to be fed back to the base station rather than the specific packets a user is missing. Furthermore, since network coding on retransmission renders each coded packet potentially useful to every user still missing degrees of freedom, the base station or transmitting node does not need to know how many degrees of freedom *each* user is missing. Rather, the base station only needs to know how many degrees of freedom it needs to send to fill-in all the erasures of the node missing the most degrees of freedom with high probability. If the base station transmits this number of coded packets, then with high probability all the remaining nodes will also have all their erasures filled in as well.

Each user can form an estimate of the erasure probability p_i associated with its channel based on how many of the incoming packets were erased in the first k transmissions. Knowing that the next batch of coded packets that will arrive are likely to experience a similar proportion of erasures, each user will scale up the number of degrees of freedom that the base station should transmit for it to decode the k packets and will send this scaled number, instead of M_i^t, to the base station. One strategy to scale M_i^t is to request $M_i^t/(1-p_i)$ packets. The scaled number $M_i^t/(1-p_i)$ is the expected number of packets that need to be transmitted in order for a receiver to successfully decode M_i^t of them, if the channel erasure probability were p_i. The base station will retransmit as many coded packets M_t as requested by the user that has experienced the highest number of erasures. Thus,

$$M_t^* = \max_{i \in [1,n]} \left(\frac{M_i^t}{1-p_i} \right) \tag{14}$$

After the base station completes the desired number of retransmissions, another feedback slot is allocated and the process is repeated until every user has received and decoded all packets. For each such round, note that only a single time slot is devoted to the acknowledgment process for n users. Therefore, at most several packets of feedback will be needed.

B. Generalization of Jitter Mechanism

In order to enhance the robustness of using the location of a pulse or series of pulses within a time slot to indicate the number of degrees of freedom that need to be transmitted, codewords can be employed within these pulses. The choice of codeword should depend on how the channel is modeled at the bit level (i.e. choose the bits that make up the codeword such that the codeword has very high fidelity). The use of multi-bit codewords will reduce the probability of error in the feedback, which is important when the position of the error can make a significant difference in the penalty incurred for that error. For example, consider a scenario in which the feedback slot has 3 subslots and a pulse in the first, second, and third subslot will indicate a loss of 100, 50, and 10 packets respectively. An error in the first subslot could result in 90 extra transmissions from the base station which is not the same penalty as if the error had occurred in the second or third subslots. A solution for this high variance in penalty could be to have codewords of different lengths for different subslots.

The above discussion raises the questions of the optimal number of subslots and the correspondence between subslot location and degrees of freedom needed. Since responses from different users will all be transmitted in the single time slot, there could potentially be multiple access interference, or misinterpretation of degrees of freedom requested, if the users are not synchronized with each other and the base station. However, the base station only needs to receive M_t, as defined in equation (14), thus we propose the following scheme: the larger the number of degrees of freedom a receiving node will request, the earlier the subslot in which it will transmit within the single feedback slot. Thus, the base station will aim to find the first subslot in which a user transmits a codeword. Even with coarse synchronization, the base station can detect the first subslot in which it receives considerable energy, even if it can not decode the codeword, and at worst, it transmits more dof than needed. Better synchronization will enable more bandwidth savings so that only M_t dof will be transmitted: for optimal performance the precision (granularity) of the

synchronization should correspond to the length in time of a bit, so as to allow decoding of codewords. The subslot size can be chosen to accommodate the synchronization capability, and a decreased synchronization ability will result in coarser feedback, so that the otherwise extremely large bandwidth savings in the acknowledgment and retransmission process might be slightly reduced.

The optimal number of subslots in the feedback slot should be determined based on $\hat{p_i}$, k and n. As an example, consider a feedback packet of size 500 bits, and note that we could in principle indicate 500 distinct feedback levels, but that would make the feedback more susceptible to noise and synchronization errors. We could select 50 subslots per time slot, a choice which would allow a 10 bit codeword for each subslot. Also note that when p_i is small, the probability that a user has missed a significant portion of the packets is very small and it might be wasteful to have many subslots that indicate such large numbers. Finally, a protocol that allocates these subslots based on a percentage might be more universal and can be implemented without changing any of the underlying physical layer elements. Under such protocols, a feedback pulse in the last subslot could indicate a loss of less than $\left(\frac{1}{n}\right)^{th}$ of the packets and in the second to last subslot could indicate twice that amount and so on.

Another enhancement that could reduce the number of users that would participate in the feedback is the use of an auction message from the base station. As introduced by Peters et. al [11], the principle idea is that of sending an "auction message" from the transmitting node to state what it thinks the state of the nodes are. Nodes only send back NACKs if the auction message differs significantly from their actual state. Sending such auction messages out when there is only a small probability that all files are complete can still yield great savings with the single slot NACK.

C. Generalization to CDMA Code Selection

The pulse position indication described in the previous two subsections is one example of a CDMA code. Another example is direct sequence spread spectrum (DSSS), which can also be used to communicate how many degrees of freedom a node needs. More generally, any type of CDMA code can be used, and the choice of a particular CDMA code transmitted by each node can correspond to the range of numbers of or percentage of degrees of freedom, such as $M_i^t/(1-p_i)$, that the node is requesting from the base station. For example, if DSSS were used, then the base station would first apply the matched filter corresponding to the highest percentage range of dofs requested. If a detection is found, the base station would be done processing the NACK slot, and would then transmit the highest number of dofs. If a detection is not found, the base station would next apply the matched filter corresponding to the second highest number of dofs, and the process is repeated. The CDMA codes can be selected so that the correlation between codes corresponding to adjacent numbers of dofs (for example the highest number and the second highest) to be retransmitted is higher than the correlation between codes representing a large difference in the numbers of degrees of freedom to be retransmitted. This selection choice would increase robustness of the protocol to errors in detection of the single slot NACK.

V. CONCLUSION

In a wireless broadcast scenario in which a peer node transmits coded packets to independent users, a feedback mechanism is required to notify the transmitting node if all packets are received by every user or if further transmissions are required. In this paper we presented a predictive model to determine the optimal feedback time in a broadcast erasure channel that will reduce the feedback traffic. We investigated the scalability of our model for increasing file sizes, varying channel erasure probabilities, and most notably large number of receivers. We analytically showed that the number of time slots needed to reliably transmit a file to n users increases as log of the number of receivers. We examined the robustness of this model to wrong channel estimation and lack of knowledge about the number of receivers.

We also introduced a new *single slot* feedback mechanism, that enables multiple receivers to give their feedback simultaneously, and gave a general framework for its implementation. We noted the attempts made by others to reduce feedback traffic, specifically we discussed the performance of NACK-Oriented Reliable Multicast (NORM) protocol enhanced by timer based back-off mechanism. However, none of these methods would reduce the amount of feedback needed by nearly as much as the factor of $n \times k$ savings of our method, which consolidates and enables feedback for *all packets from all users* into a *single* time slot.

REFERENCES

[1] B. Adamson, C. Bormann, M. Handley, and J. Macker, "NACK-Oriented Reliable Multicast (NORM) Transport Protocol," *Internet Engineering Task Force (IETF) RFC*, vol. 5740, 2009.

[2] D. Lucani, "Network coding for delay challenged environments," 2010.

[3] R. Adamson and J. Macker, "Quantitative prediction of nack-oriented reliable multicast (norm) feedback," in *MILCOM 2002. Proceedings*, vol. 2, 2002, pp. 964 – 969 vol.2.

[4] F. Steutel and J. Thiemann, *The gamma process and the Poisson distribution*. Dept. of Mathematics and Computing Science, Univ. of Technology, 1989.

[5] H. Harter, "Expected values of normal order statistics," *Biometrika*, vol. 48, no. 1-2, p. 151, 1961.

[6] A. Sarhan and B. Greenberg, *Contributions to order statistics*. A Wiley Publication in Applied Statistics, 1962.

[7] D. Sinha, H. Zhou, and N. Shenoy, "Advances in computation of the maximum of a set of Gaussian random variables," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 26, no. 8, pp. 1522–1533, 2007.

[8] P. Strait, "An identity on the maximum of a set of random variables," *Journal of Multivariate Analysis*, vol. 4, no. 4, pp. 494–496, 1974.

[9] K. Briggs, L. Song, and T. Prellberg, "A note on the distribution of the maximum of a set of Poisson random variables," *Arxiv preprint arXiv:0903.4373*, 2009.

[10] A. Eryilmaz, A. Ozdaglar, and M. Médard, "On delay performance gains from network coding," in *Information Sciences and Systems, 2006 40th Annual Conference on*, 2006, pp. 864 –870.

[11] A. Peters, A. Rezaee, L. Zeger, and M. Médard, "New techniques to prevent ack explosion in multicast," *Invention Disclosure*, 2010.